```
import numpy as np
import matplotlib.pyplot as plt
from scipy.ndimage import gaussian_filter, map_coordinates
from scipy.ndimage import shift
from scipy.ndimage import rotate

def compute_iou(mask1, mask2):
    # Ensure binary values (0 or 1)
    mask1 = (mask1 > 0.5).astype(np.uint8)  # Threshold at 0.5
    mask2 = (mask2 > 0.5).astype(np.uint8)

    # Compute intersection and union
    intersection = np.logical_and(mask1, mask2).sum()
    union = np.logical_or(mask1, mask2).sum()

    # Avoid division by zero
    iou = intersection / union if union > 0 else 0.0
    return iou # 1- iou # for GA optimization, we need to minimize. So it should be 1- iou


def ImageTranslateRotate(img, tx, rz):
  # Define translation amounts (positive shifts right/down, negative shifts left/up)
  shift_x = tx  # Shift right by 10 pixels
  shift_y = 0  # Shift up by 5 pixels

  # Apply translation
  translated_image = shift(img, shift=(shift_y, shift_x), mode='nearest')

  # Define the rotation angle (in degrees)
  rotation_angle = rz  # Rotate 30 degrees counterclockwise

  # Apply rotation
  rotated_image = rotate(translated_image, angle=rotation_angle, reshape=False, mode='nearest')

  return rotated_image
```

```
import numpy as np
import matplotlib.pyplot as plt
import random
from tabulate import tabulate
import copy



# objective function
def ObjectiveFunction(x1,x2):
    image = np.load('/content/drive/MyDrive/SCAPro/org_image.npy')
    deformed_image = np.load('/content/drive/MyDrive/SCAPro/Gr3.npy')
    deformed_image = ImageTranslateRotate(deformed_image, x1, x2)
    y = 1 - compute_iou(image, deformed_image)
    plt.imshow(image, cmap='gray')
    plt.imshow(deformed_image, cmap='turbo', alpha = 0.5)
    return y

def GetRandomBinary(chromolength):
    p1 =''
    for i in range(chromolength):
        p1 = p1+str(random.randint(0, 1))

    return p1

def SolutionRepresentation(nump,chromolength):
    population = []
    for i in range(nump):
        population.append(GetRandomBinary(chromolength))

    return population


def GetFittnessTable(population,x1L, x1U, x2L, x2U, chromolength,printtable = False, plot = False):
    fitness = []
    for i in range(len(population)):
        x1c = population[i][:int(chromolength/2)]
        x2c = population[i][int(chromolength/2):]

        x1d = int(x1c, 2)
        x2d = int(x2c, 2)

        x1 = x1L + ( (x1U-x1L)/(2**(chromolength/2)-1) ) * x1d
        x2 = x2L + ( (x2U-x2L)/(2**(chromolength/2)-1) ) * x2d
        x2 = x2L + ( (x2U-x2L)/(2**(chromolength/2)-1) ) * x2d

        objval = ObjectiveFunction(x1,x2)
        if plot:
          plt.show()
        fitness.append([i,population[i],x1c,x2c,x1d,x2d,x1,x2,objval])

    fitness = np.asarray(fitness)
    if printtable == True:
        # Generate the table in fancy format.
        headers = ['Index','population','x1c','x2c','x1d','x2d','x1','x2','objval']
        table = tabulate(fitness, headers, tablefmt="fancy_grid")
        print(table)

    return fitness


# TournamentSelection
def TournamentSelection(fitnesstable,populationsize, printtable=False):
    numtournament = 2
    newpopulation =[]
    for i in range(numtournament):
        index = np.array(fitnesstable[:,0],dtype=np.uint8)
        sampled_list = np.random.choice(index, size=(int(populationsize/2), 2),replace=False)
        print('tournament list \n',sampled_list)

        objvalue = np.array(fitnesstable[:,8],dtype=np.float64)

        for si in range(int(populationsize/2)):
            if objvalue[sampled_list[si,0]] < objvalue[sampled_list[si,1]]:
                betterindex = sampled_list[si,0]
            else:
                betterindex = sampled_list[si,1]

            newpopulation.append(fitnesstable[betterindex,:])

    newpopulation = np.array(newpopulation)

    if printtable == True:
        # Generate the table in fancy format.
        headers = ['Index','population','x1c','x2c','x1d','x2d','x1','x2','objval']
        table = tabulate(newpopulation, headers, tablefmt="fancy_grid")
        print('TournamentSelection')
        print(table)

    return newpopulation



# Crossover
def Crossover(p1, p2):
    assert(len(p1) == len(p2))

    site = np.random.randint(1,len(p1), size=1)[0]
    print('crossover site: ',site)
    p1t = p1[:site]+p2[site:]
    p2t = p2[:site]+p1[site:]

    return p1t, p2t

def GetCrossoverTable(table,populationsize,x1L, x1U, x2L, x2U,chromolength, printtable=False):
    probcrossover = 0.9

    index = np.array(table[:,0],dtype=np.uint8)
    sampled_list = np.random.choice(index, size=(int(populationsize/2), 2),replace=False)
    print('Crossover list \n',sampled_list)
    population = table[:,1]

    newpopulation =[]
    for si in range(int(populationsize/2)):
        p1, p2 = population[sampled_list[si,0]], population[sampled_list[si,1]]

        prob = random.random()
        if prob<probcrossover:
            print('prob {} < probcrossover of {}. Perform crossover'.format(prob,probcrossover))
            o1, o2 = Crossover(p1, p2)
        else:
            print('prob {} > probcrossover of {}. DO NOT Perform crossover'.format(prob,probcrossover))
            o1, o2 = p1, p2

        x1c = o1[:int(chromolength/2)]
        x2c = o1[int(chromolength/2):]
        x1d = int(x1c, 2)
        x2d = int(x2c, 2)
        x1 = x1L + ( (x1U-x1L)/(2**(chromolength/2)-1) ) * x1d
        x2 = x2L + ( (x2U-x2L)/(2**(chromolength/2)-1) ) * x2d
        objval = ObjectiveFunction(x1,x2)
        newpopulation.append([sampled_list[si,0],o1,x1c,x2c,x1d,x2d,x1,x2,objval])

        x1c = o2[:int(chromolength/2)]
        x2c = o2[int(chromolength/2):]
        x1d = int(x1c, 2)
        x2d = int(x2c, 2)
        x1 = x1L + ( (x1U-x1L)/(2**(chromolength/2)-1) ) * x1d
        x2 = x2L + ( (x2U-x2L)/(2**(chromolength/2)-1) ) * x2d
        objval = ObjectiveFunction(x1,x2)
        newpopulation.append([sampled_list[si,1],o2,x1c,x2c,x1d,x2d,x1,x2,objval])

    newpopulation = np.array(newpopulation)

    if printtable == True:
        # Generate the table in fancy format.
        headers = ['Index','population','x1c','x2c','x1d','x2d','x1','x2','objval']
        table = tabulate(newpopulation, headers, tablefmt="fancy_grid")
        print('CrossoverTable')
        print(table)

    return newpopulation


# mutation
def Mutation(p1):
    site = np.random.randint(0,len(p1), size=1)[0]
    print('mutation site: ',site)

    if p1[site]=='1':
        p1t = p1[:site]+'0'+ p1[(site+1):]
    else:
        p1t = p1[:site]+'1'+ p1[(site+1):]

    return p1t

def GetMutationTable(table,populationsize,x1L, x1U, x2L, x2U,chromolength, printtable=False):
    probmutation = 0.1

    index = np.array(table[:,0],dtype=np.uint8)
    population = table[:,1]

    newpopulation =[]
    for si in range(populationsize):
        p1 = population[si]

        prob = random.random()
        if prob<probmutation:
            print('prob {} < probmutation of {}. Perform Mutation'.format(prob,probmutation))
            o1 = Mutation(p1)
        else:
            print('prob {} > probmutation of {}. DO NOT Perform Mutation'.format(prob,probmutation))
            o1 = p1

        x1c = o1[:int(chromolength/2)]
        x2c = o1[int(chromolength/2):]
        x1d = int(x1c, 2)
        x2d = int(x2c, 2)
        x1 = x1L + ( (x1U-x1L)/(2**(chromolength/2)-1) ) * x1d
        x2 = x2L + ( (x2U-x2L)/(2**(chromolength/2)-1) ) * x2d
        objval = ObjectiveFunction(x1,x2)
        newpopulation.append([si,o1,x1c,x2c,x1d,x2d,x1,x2,objval])

    newpopulation = np.array(newpopulation)

    if printtable == True:
        # Generate the table in fancy format.
        headers = ['Index','population','x1c','x2c','x1d','x2d','x1','x2','objval']
        table = tabulate(newpopulation, headers, tablefmt="fancy_grid")
        print('MutationTable')
        print(table)

    return newpopulation


def GetSurvivalTable(oldpop,newpop,populationsize, printtable=False):
    combine = np.concatenate((oldpop,newpop),axis=0)
    objval = np.array(combine[:,8],dtype=np.float64)

    newpopulation = []
    for i in range(populationsize):
        minindex = np.argmin(objval)
        objval[minindex] = objval[minindex] + 1e6
        newpopulation.append(combine[minindex,:])

    newpopulation = np.array(newpopulation)

    if printtable == True:
        # Generate the table in fancy format.
        headers = ['Index','population','x1c','x2c','x1d','x2d','x1','x2','objval']
        table = tabulate(newpopulation, headers, tablefmt="fancy_grid")
        print('SurvivalTable')
        print(table)

    return newpopulation


# user input ===================================================================================
chromolength = 10
populationsize = 8
maxgeneration  = 5

x1L, x1U = -5, -2
x2L, x2U = 8, 10


assert(chromolength%2 == 0)
assert(populationsize%2 == 0)


solrep = SolutionRepresentation(populationsize,chromolength)
print('Initial population \n',solrep)


print('Fitness table for initial population \n')
oldpopulation = GetFittnessTable(solrep,x1L, x1U, x2L, x2U, chromolength, printtable = True)


pointsforplot = np.array(oldpopulation[:,6:],dtype=np.float64)

genitr =1
while genitr < maxgeneration:
    newpopulation = TournamentSelection(oldpopulation,populationsize,printtable=True)
    newpopulation[:,0]= [i for i in range(populationsize)]

    newpopulation2 = GetCrossoverTable(newpopulation,populationsize,x1L, x1U, x2L, x2U,chromolength, printtable=True)
    newpopulation2[:,0]= [i for i in range(populationsize)]

    newpopulation3 = GetMutationTable(newpopulation2,populationsize,x1L, x1U, x2L, x2U,chromolength, printtable=True)
    newpopulation3[:,0]= [i for i in range(populationsize)]

    newpopulation4 = GetSurvivalTable(newpopulation3,oldpopulation,populationsize, printtable=True)
    newpopulation4[:,0]= [i for i in range(populationsize)]

    oldpopulation = copy.deepcopy(newpopulation4)

    pointsforplot = np.array(oldpopulation[:,6:],dtype=np.float64)
    genitr = genitr + 1



solrep = oldpopulation[:,1]
print('Final result')
oldpopulation = GetFittnessTable(solrep,x1L, x1U, x2L, x2U, chromolength, printtable = True, plot = True)
```

```
Initial population
 ['1001111000', '1100011001', '0000010100', '0011001001', '1001010111', '1111011001', '0101111001', '1011010011']
Fitness table for initial population
```

Index	population	x1c	x2c	x1d	x2d	x1	x2	objval
0	1001111000	10011	11000	19	24	-3.16129	9.54839	0.0798077
1	1100011001	11000	11001	24	25	-2.67742	9.6129	0.0655577
2	0000010100	00000	10100	0	20	-5	9.29032	0.341484
3	0011001001	00110	01001	6	9	-4.41935	8.58065	0.253076
4	1001010111	10010	10111	18	23	-3.25806	9.48387	0.0861244
5	1111011001	11110	11001	30	25	-2.09677	9.6129	0.137769
6	0101111001	01011	11001	11	25	-3.93548	9.6129	0.174559
7	1011010011	10110	10011	22	19	-2.87097	9.22581	0.0562685

```
tournament list
 [[1 4]
 [6 5]
 [2 3]
 [7 0]]
tournament list
 [[1 0]
 [3 6]
 [4 5]
 [7 2]]
TournamentSelection
```

Index	population	x1c	x2c	x1d	x2d	x1	x2	objval
1	1100011001	11000	11001	24	25	-2.67742	9.6129	0.0655577
5	1111011001	11110	11001	30	25	-2.09677	9.6129	0.137769
3	0011001001	00110	01001	6	9	-4.41935	8.58065	0.253076
7	1011010011	10110	10011	22	19	-2.87097	9.22581	0.0562685
1	1100011001	11000	11001	24	25	-2.67742	9.6129	0.0655577
6	0101111001	01011	11001	11	25	-3.93548	9.6129	0.174559
4	1001010111	10010	10111	18	23	-3.25806	9.48387	0.0861244
7	1011010011	10110	10011	22	19	-2.87097	9.22581	0.0562685

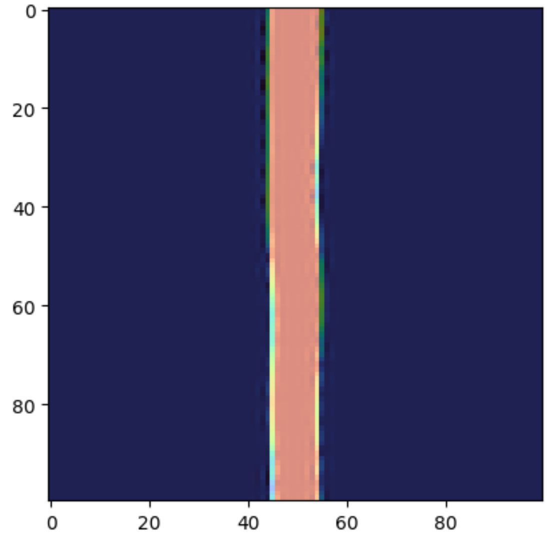
```
Crossover list
 [[2 1]
 [5 3]
 [0 7]
 [6 4]]
prob 0.8281001512180831 < probcrossover of 0.9. Perform crossover
crossover site:  4
prob 0.8309010832701005 < probcrossover of 0.9. Perform crossover
crossover site:  2
prob 0.04951808822689474 < probcrossover of 0.9. Perform crossover
crossover site:  4
prob 0.2148393617416896 < probcrossover of 0.9. Perform crossover
crossover site:  8
CrossoverTable
```

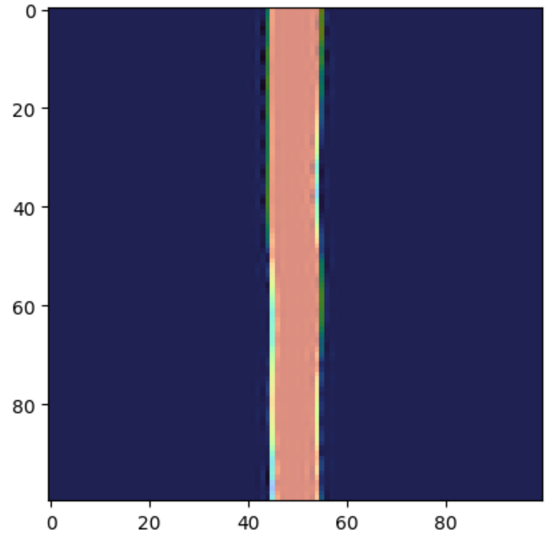
Index	population	x1c	x2c	x1d	x2d	x1	x2	objval
2	0011011001	00110	11001	6	25	-4.41935	9.6129	0.266319
1	1111001001	11110	01001	30	9	-2.09677	8.58065	0.155493
5	0111010011	01110	10011	14	19	-3.64516	9.22581	0.132834
3	1001111001	10011	11001	19	25	-3.16129	9.6129	0.0853308
0	1100010011	11000	10011	24	19	-2.67742	9.22581	0.0556098
7	1011011001	10110	11001	22	25	-2.87097	9.6129	0.0486594
6	1001010101	10010	10101	18	21	-3.25806	9.35484	0.0837344
4	1100011011	11000	11011	24	27	-2.67742	9.74194	0.0717092

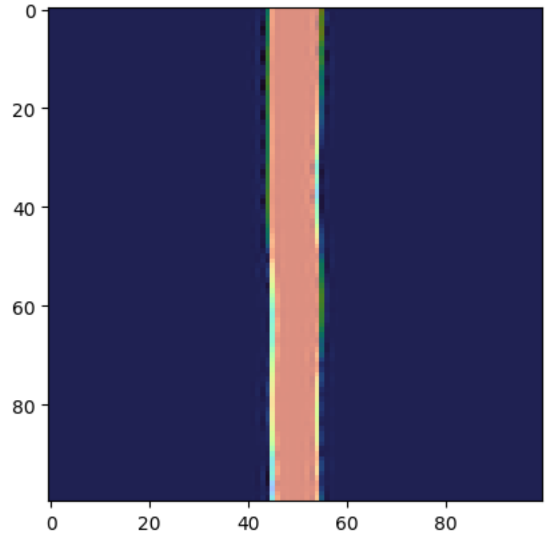
```
prob 0.013064518927627056 < probmutation of 0.1. Perform Mutation
mutation site:  3
prob 0.5912711045113261 > probmutation of 0.1. DO NOT Perform Mutation
prob 0.9305035605624258 > probmutation of 0.1. DO NOT Perform Mutation
prob 0.5751745445792998 > probmutation of 0.1. DO NOT Perform Mutation
prob 0.27287174687239446 > probmutation of 0.1. DO NOT Perform Mutation
prob 0.8550924307255684 > probmutation of 0.1. DO NOT Perform Mutation
prob 0.046200403740738305 < probmutation of 0.1. Perform Mutation
mutation site:  4
prob 0.7929771103642891 > probmutation of 0.1. DO NOT Perform Mutation
MutationTable
╒═════════╤══════════════╤═══════╤═══════╤═══════╤═══════╤══════════╤═════════╤═══════════╕
│   Index │   population │   x1c │   x2c │   x1d │   x2d │       x1 │      x2 │    objval │
╞═════════╪══════════════╪═══════╪═══════╪═══════╪═══════╪══════════╪═════════╪═══════════╡
│       0 │   0010011001 │ 00100 │ 11001 │     4 │    25 │ -4.6129  │ 9.6129  │ 0.282741  │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       1 │   1111001001 │ 11110 │ 01001 │    30 │     9 │ -2.09677 │ 8.58065 │ 0.155493  │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       2 │   0111010011 │ 01110 │ 10011 │    14 │    19 │ -3.64516 │ 9.22581 │ 0.132834  │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       3 │   1001111001 │ 10011 │ 11001 │    19 │    25 │ -3.16129 │ 9.6129  │ 0.0853308 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       4 │   1100010011 │ 11000 │ 10011 │    24 │    19 │ -2.67742 │ 9.22581 │ 0.0556098 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       5 │   1011011001 │ 10110 │ 11001 │    22 │    25 │ -2.87097 │ 9.6129  │ 0.0486594 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       6 │   1001110101 │ 10011 │ 10101 │    19 │    21 │ -3.16129 │ 9.35484 │ 0.0673828 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       7 │   1100011011 │ 11000 │ 11011 │    24 │    27 │ -2.67742 │ 9.74194 │ 0.0717092 │
╘═════════╧══════════════╧═══════╧═══════╧═══════╧═══════╧══════════╧═════════╧═══════════╛
SurvivalTable
╒═════════╤══════════════╤═══════╤═══════╤═══════╤═══════╤══════════╤═════════╤═══════════╕
│   Index │   population │   x1c │   x2c │   x1d │   x2d │       x1 │      x2 │    objval │
╞═════════╪══════════════╪═══════╪═══════╪═══════╪═══════╪══════════╪═════════╪═══════════╡
│       5 │   1011011001 │ 10110 │ 11001 │    22 │    25 │ -2.87097 │ 9.6129  │ 0.0486594 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       4 │   1100010011 │ 11000 │ 10011 │    24 │    19 │ -2.67742 │ 9.22581 │ 0.0556098 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       7 │   1011010011 │ 10110 │ 10011 │    22 │    19 │ -2.87097 │ 9.22581 │ 0.0562685 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       1 │   1100011001 │ 11000 │ 11001 │    24 │    25 │ -2.67742 │ 9.6129  │ 0.0655577 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       6 │   1001110101 │ 10011 │ 10101 │    19 │    21 │ -3.16129 │ 9.35484 │ 0.0673828 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       7 │   1100011011 │ 11000 │ 11011 │    24 │    27 │ -2.67742 │ 9.74194 │ 0.0717092 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       0 │   1001111000 │ 10011 │ 11000 │    19 │    24 │ -3.16129 │ 9.54839 │ 0.0798077 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       3 │   1001111001 │ 10011 │ 11001 │    19 │    25 │ -3.16129 │ 9.6129  │ 0.0853308 │
╘═════════╧══════════════╧═══════╧═══════╧═══════╧═══════╧══════════╧═════════╧═══════════╛
tournament list
 [[4 2]
 [1 5]
 [3 6]
 [0 7]]
tournament list
 [[4 6]
 [5 0]
 [1 7]
 [3 2]]
TournamentSelection
╒═════════╤══════════════╤═══════╤═══════╤═══════╤═══════╤══════════╤═════════╤═══════════╕
│   Index │   population │   x1c │   x2c │   x1d │   x2d │       x1 │      x2 │    objval │
╞═════════╪══════════════╪═══════╪═══════╪═══════╪═══════╪══════════╪═════════╪═══════════╡
│       2 │   1011010011 │ 10110 │ 10011 │    22 │    19 │ -2.87097 │ 9.22581 │ 0.0562685 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       1 │   1100010011 │ 11000 │ 10011 │    24 │    19 │ -2.67742 │ 9.22581 │ 0.0556098 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       3 │   1100011001 │ 11000 │ 11001 │    24 │    25 │ -2.67742 │ 9.6129  │ 0.0655577 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       0 │   1011011001 │ 10110 │ 11001 │    22 │    25 │ -2.87097 │ 9.6129  │ 0.0486594 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       4 │   1001110101 │ 10011 │ 10101 │    19 │    21 │ -3.16129 │ 9.35484 │ 0.0673828 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       0 │   1011011001 │ 10110 │ 11001 │    22 │    25 │ -2.87097 │ 9.6129  │ 0.0486594 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       1 │   1100010011 │ 11000 │ 10011 │    24 │    19 │ -2.67742 │ 9.22581 │ 0.0556098 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       2 │   1011010011 │ 10110 │ 10011 │    22 │    19 │ -2.87097 │ 9.22581 │ 0.0562685 │
╘═════════╧══════════════╧═══════╧═══════╧═══════╧═══════╧══════════╧═════════╧═══════════╛
Crossover list
 [[3 0]
 [7 1]
 [2 6]
 [4 5]]
prob 0.840161148920121 < probcrossover of 0.9. Perform crossover
crossover site:  8
prob 0.6555102705584807 < probcrossover of 0.9. Perform crossover
crossover site:  3
prob 0.5031015802917279 < probcrossover of 0.9. Perform crossover
crossover site:  6
prob 0.9073137059230942 > probcrossover of 0.9. DO NOT Perform crossover
CrossoverTable
╒═════════╤══════════════╤═══════╤═══════╤═══════╤═══════╤══════════╤═════════╤═══════════╕
│   Index │   population │   x1c │   x2c │   x1d │   x2d │       x1 │      x2 │    objval │
│   Index │   population │   x1c │   x2c │   x1d │   x2d │         x1 │        x2 │    objval │
╞═════════╪══════════════╪═══════╪═══════╪═══════╪═══════╪════════════╪═══════════╪═══════════╡
│       3 │   1011011011 │ 10110 │ 11011 │    22 │    27 │   -2.87097 │ 9.74194   │ 0.0517413 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼────────────┼───────────┼───────────┤
│       0 │   1011010001 │ 10110 │ 10001 │    22 │    17 │   -2.87097 │ 9.09677   │ 0.0589971 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼────────────┼───────────┼───────────┤
│       7 │   1010010011 │ 10100 │ 10011 │    20 │    19 │   -3.06452 │ 9.22581   │ 0.0581281 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼────────────┼───────────┼───────────┤
│       1 │   1101010011 │ 11010 │ 10011 │    26 │    19 │   -2.48387 │ 9.22581   │ 0.0798077 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼────────────┼───────────┼───────────┤
│       2 │   1100010011 │ 11000 │ 10011 │    24 │    19 │   -2.67742 │ 9.22581   │ 0.0556098 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼────────────┼───────────┼───────────┤
│       6 │   1100011001 │ 11000 │ 11001 │    24 │    25 │   -2.67742 │ 9.6129    │ 0.0655577 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼────────────┼───────────┼───────────┤
│       4 │   1001110101 │ 10011 │ 10101 │    19 │    21 │   -3.16129 │ 9.35484   │ 0.0673828 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼────────────┼───────────┼───────────┤
│       5 │   1011011001 │ 10110 │ 11001 │    22 │    25 │   -2.87097 │ 9.6129    │ 0.0486594 │
╘═════════╧══════════════╧═══════╧═══════╧═══════╧═══════╧════════════╧═══════════╧═══════════╛
prob 0.162993877634257 > probmutation of 0.1. DO NOT Perform Mutation
prob 0.22927194899706882 > probmutation of 0.1. DO NOT Perform Mutation
prob 0.8514349926001227 > probmutation of 0.1. DO NOT Perform Mutation
prob 0.9469706448765063 > probmutation of 0.1. DO NOT Perform Mutation
prob 0.7147374332195803 > probmutation of 0.1. DO NOT Perform Mutation
prob 0.64584263095981 > probmutation of 0.1. DO NOT Perform Mutation
prob 0.8694643472231725 > probmutation of 0.1. DO NOT Perform Mutation
prob 0.07620189307029412 < probmutation of 0.1. Perform Mutation
mutation site:  1
MutationTable
╒═════════╤══════════════╤═══════╤═══════╤═══════╤═══════╤════════════╤═══════════╤═══════════╕
│   Index │   population │   x1c │   x2c │   x1d │   x2d │         x1 │        x2 │    objval │
╞═════════╪══════════════╪═══════╪═══════╪═══════╪═══════╪════════════╪═══════════╪═══════════╡
│       0 │   1011011011 │ 10110 │ 11011 │    22 │    27 │   -2.87097 │ 9.74194   │ 0.0517413 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼────────────┼───────────┼───────────┤
│       1 │   1011010001 │ 10110 │ 10001 │    22 │    17 │   -2.87097 │ 9.09677   │ 0.0589971 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼────────────┼───────────┼───────────┤
│       2 │   1010010011 │ 10100 │ 10011 │    20 │    19 │   -3.06452 │ 9.22581   │ 0.0581281 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼────────────┼───────────┼───────────┤
│       3 │   1101010011 │ 11010 │ 10011 │    26 │    19 │   -2.48387 │ 9.22581   │ 0.0798077 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼────────────┼───────────┼───────────┤
│       4 │   1100010011 │ 11000 │ 10011 │    24 │    19 │   -2.67742 │ 9.22581   │ 0.0556098 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼────────────┼───────────┼───────────┤
│       5 │   1100011001 │ 11000 │ 11001 │    24 │    25 │   -2.67742 │ 9.6129    │ 0.0655577 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼────────────┼───────────┼───────────┤
│       6 │   1001110101 │ 10011 │ 10101 │    19 │    21 │   -3.16129 │ 9.35484   │ 0.0673828 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼────────────┼───────────┼───────────┤
│       7 │   1111011001 │ 11110 │ 11001 │    30 │    25 │   -2.09677 │ 9.6129    │ 0.137769  │
╘═════════╧══════════════╧═══════╧═══════╧═══════╧═══════╧════════════╧═══════════╧═══════════╛
SurvivalTable
╒═════════╤══════════════╤═══════╤═══════╤═══════╤═══════╤════════════╤═══════════╤═══════════╕
│   Index │   population │   x1c │   x2c │   x1d │   x2d │         x1 │        x2 │    objval │
╞═════════╪══════════════╪═══════╪═══════╪═══════╪═══════╪════════════╪═══════════╪═══════════╡
│       0 │   1011011001 │ 10110 │ 11001 │    22 │    25 │   -2.87097 │ 9.6129    │ 0.0486594 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼────────────┼───────────┼───────────┤
│       0 │   1011011011 │ 10110 │ 11011 │    22 │    27 │   -2.87097 │ 9.74194   │ 0.0517413 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼────────────┼───────────┼───────────┤
│       4 │   1100010011 │ 11000 │ 10011 │    24 │    19 │   -2.67742 │ 9.22581   │ 0.0556098 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼────────────┼───────────┼───────────┤
│       1 │   1100010011 │ 11000 │ 10011 │    24 │    19 │   -2.67742 │ 9.22581   │ 0.0556098 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼────────────┼───────────┼───────────┤
│       2 │   1011010011 │ 10110 │ 10011 │    22 │    19 │   -2.87097 │ 9.22581   │ 0.0562685 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼────────────┼───────────┼───────────┤
│       2 │   1010010011 │ 10100 │ 10011 │    20 │    19 │   -3.06452 │ 9.22581   │ 0.0581281 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼────────────┼───────────┼───────────┤
│       1 │   1011010001 │ 10110 │ 10001 │    22 │    17 │   -2.87097 │ 9.09677   │ 0.0589971 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼────────────┼───────────┼───────────┤
│       5 │   1100011001 │ 11000 │ 11001 │    24 │    25 │   -2.67742 │ 9.6129    │ 0.0655577 │
╘═════════╧══════════════╧═══════╧═══════╧═══════╧═══════╧════════════╧═══════════╧═══════════╛
tournament list
 [[3 7]
 [0 1]
 [6 5]
 [4 2]]
tournament list
 [[5 2]
 [1 4]
 [0 6]
 [7 3]]
TournamentSelection
╒═════════╤══════════════╤═══════╤═══════╤═══════╤═══════╤════════════╤═══════════╤═══════════╕
│   Index │   population │   x1c │   x2c │   x1d │   x2d │         x1 │        x2 │    objval │
╞═════════╪══════════════╪═══════╪═══════╪═══════╪═══════╪════════════╪═══════════╪═══════════╡
│       3 │   1100010011 │ 11000 │ 10011 │    24 │    19 │   -2.67742 │ 9.22581   │ 0.0556098 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼────────────┼───────────┼───────────┤
│       0 │   1011011001 │ 10110 │ 11001 │    22 │    25 │   -2.87097 │ 9.6129    │ 0.0486594 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼────────────┼───────────┼───────────┤
│       5 │   1010010011 │ 10100 │ 10011 │    20 │    19 │   -3.06452 │ 9.22581   │ 0.0581281 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼────────────┼───────────┼───────────┤
│       2 │   1100010011 │ 11000 │ 10011 │    24 │    19 │   -2.67742 │ 9.22581   │ 0.0556098 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼────────────┼───────────┼───────────┤
│       2 │   1100010011 │ 11000 │ 10011 │    24 │    19 │   -2.67742 │ 9.22581   │ 0.0556098 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼────────────┼───────────┼───────────┤
```

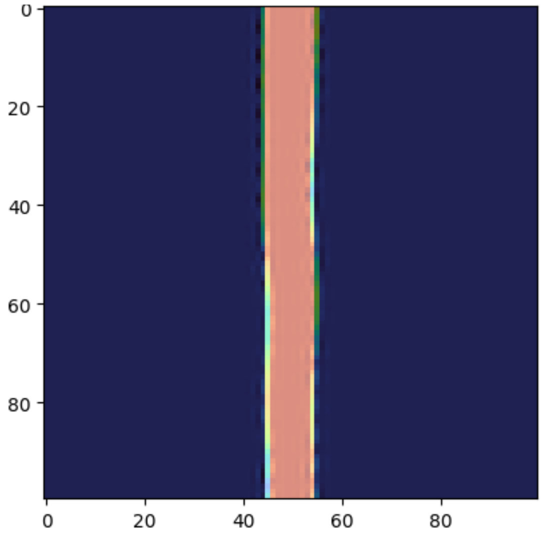
```
│       1 │   1011011011 │ 10110 │ 11011 │    22 │    27 │ -2.87097 │ 9.74194 │ 0.0517413 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       0 │   1011011001 │ 10110 │ 11001 │    22 │    25 │ -2.87097 │ 9.6129  │ 0.0486594 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       3 │   1100010011 │ 11000 │ 10011 │    24 │    19 │ -2.67742 │ 9.22581 │ 0.0556098 │
╘═════════╧══════════════╧═══════╧═══════╧═══════╧═══════╧══════════╧═════════╧═══════════╛
Crossover list
 [[3 1]
 [6 0]
 [4 7]
 [2 5]]
prob 0.931981981977063 > probcrossover of 0.9. DO NOT Perform crossover
prob 0.11820742254045424 < probcrossover of 0.9. Perform crossover
crossover site:  9
prob 0.14713846693286503 < probcrossover of 0.9. Perform crossover
crossover site:  9
prob 0.6362550866593426 < probcrossover of 0.9. Perform crossover
crossover site:  6
CrossoverTable
╒═════════╤══════════════╤═══════╤═══════╤═══════╤═══════╤══════════╤═════════╤═══════════╕
│   Index │   population │   x1c │   x2c │   x1d │   x2d │       x1 │      x2 │    objval │
╞═════════╪══════════════╪═══════╪═══════╪═══════╪═══════╪══════════╪═════════╪═══════════╡
│       3 │   1100010011 │ 11000 │ 10011 │    24 │    19 │ -2.67742 │ 9.22581 │ 0.0556098 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       1 │   1011011001 │ 10110 │ 11001 │    22 │    25 │ -2.87097 │ 9.6129  │ 0.0486594 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       6 │   1011011001 │ 10110 │ 11001 │    22 │    25 │ -2.87097 │ 9.6129  │ 0.0486594 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       0 │   1100010011 │ 11000 │ 10011 │    24 │    19 │ -2.67742 │ 9.22581 │ 0.0556098 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       4 │   1100010011 │ 11000 │ 10011 │    24 │    19 │ -2.67742 │ 9.22581 │ 0.0556098 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       7 │   1100010011 │ 11000 │ 10011 │    24 │    19 │ -2.67742 │ 9.22581 │ 0.0556098 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       2 │   1010011011 │ 10100 │ 11011 │    20 │    27 │ -3.06452 │ 9.74194 │ 0.084372  │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       5 │   1011010011 │ 10110 │ 10011 │    22 │    19 │ -2.87097 │ 9.22581 │ 0.0562685 │
╘═════════╧══════════════╧═══════╧═══════╧═══════╧═══════╧══════════╧═════════╧═══════════╛
prob 0.11969606321387039 > probmutation of 0.1. DO NOT Perform Mutation
prob 0.9212746995286559 > probmutation of 0.1. DO NOT Perform Mutation
prob 0.7205904898024131 > probmutation of 0.1. DO NOT Perform Mutation
prob 0.8791004627180958 > probmutation of 0.1. DO NOT Perform Mutation
prob 0.024686921867449474 < probmutation of 0.1. Perform Mutation
mutation site:  3
prob 0.23516200399368992 > probmutation of 0.1. DO NOT Perform Mutation
prob 0.5747248037625975 > probmutation of 0.1. DO NOT Perform Mutation
prob 0.9636401763836688 > probmutation of 0.1. DO NOT Perform Mutation
MutationTable
╒═════════╤══════════════╤═══════╤═══════╤═══════╤═══════╤══════════╤═════════╤═══════════╕
│   Index │   population │   x1c │   x2c │   x1d │   x2d │       x1 │      x2 │    objval │
╞═════════╪══════════════╪═══════╪═══════╪═══════╪═══════╪══════════╪═════════╪═══════════╡
│       0 │   1100010011 │ 11000 │ 10011 │    24 │    19 │ -2.67742 │ 9.22581 │ 0.0556098 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       1 │   1011011001 │ 10110 │ 11001 │    22 │    25 │ -2.87097 │ 9.6129  │ 0.0486594 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       2 │   1011011001 │ 10110 │ 11001 │    22 │    25 │ -2.87097 │ 9.6129  │ 0.0486594 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       3 │   1100010011 │ 11000 │ 10011 │    24 │    19 │ -2.67742 │ 9.22581 │ 0.0556098 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       4 │   1101010011 │ 11010 │ 10011 │    26 │    19 │ -2.48387 │ 9.22581 │ 0.0798077 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       5 │   1100010011 │ 11000 │ 10011 │    24 │    19 │ -2.67742 │ 9.22581 │ 0.0556098 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       6 │   1010011011 │ 10100 │ 11011 │    20 │    27 │ -3.06452 │ 9.74194 │ 0.084372  │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       7 │   1011010011 │ 10110 │ 10011 │    22 │    19 │ -2.87097 │ 9.22581 │ 0.0562685 │
╘═════════╧══════════════╧═══════╧═══════╧═══════╧═══════╧══════════╧═════════╧═══════════╛
SurvivalTable
╒═════════╤══════════════╤═══════╤═══════╤═══════╤═══════╤══════════╤═════════╤═══════════╕
│   Index │   population │   x1c │   x2c │   x1d │   x2d │       x1 │      x2 │    objval │
╞═════════╪══════════════╪═══════╪═══════╪═══════╪═══════╪══════════╪═════════╪═══════════╡
│       1 │   1011011001 │ 10110 │ 11001 │    22 │    25 │ -2.87097 │ 9.6129  │ 0.0486594 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       2 │   1011011001 │ 10110 │ 11001 │    22 │    25 │ -2.87097 │ 9.6129  │ 0.0486594 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       0 │   1011011001 │ 10110 │ 11001 │    22 │    25 │ -2.87097 │ 9.6129  │ 0.0486594 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       1 │   1011011011 │ 10110 │ 11011 │    22 │    27 │ -2.87097 │ 9.74194 │ 0.0517413 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       0 │   1100010011 │ 11000 │ 10011 │    24 │    19 │ -2.67742 │ 9.22581 │ 0.0556098 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       3 │   1100010011 │ 11000 │ 10011 │    24 │    19 │ -2.67742 │ 9.22581 │ 0.0556098 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       5 │   1100010011 │ 11000 │ 10011 │    24 │    19 │ -2.67742 │ 9.22581 │ 0.0556098 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       2 │   1100010011 │ 11000 │ 10011 │    24 │    19 │ -2.67742 │ 9.22581 │ 0.0556098 │
╘═════════╧══════════════╧═══════╧═══════╧═══════╧═══════╧══════════╧═════════╧═══════════╛
tournament list
 [[5 0]
 [1 7]
 [3 2]
 [5 2]
 [6 4]]
tournament list
 [[4 5]
 [0 1]
 [3 2]
 [6 7]]
TournamentSelection
╒═════════╤══════════════╤═══════╤═══════╤═══════╤═══════╤══════════╤═════════╤═══════════╕
│   Index │   population │   x1c │   x2c │   x1d │   x2d │       x1 │ x2      │    objval │
╞═════════╪══════════════╪═══════╪═══════╪═══════╪═══════╪══════════╪═════════╪═══════════╡
│       0 │   1011011001 │ 10110 │ 11001 │    22 │    25 │ -2.87097 │ 9.6129  │ 0.0486594 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       1 │   1011011001 │ 10110 │ 11001 │    22 │    25 │ -2.87097 │ 9.6129  │ 0.0486594 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       2 │   1011011001 │ 10110 │ 11001 │    22 │    25 │ -2.87097 │ 9.6129  │ 0.0486594 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       4 │   1100010011 │ 11000 │ 10011 │    24 │    19 │ -2.67742 │ 9.22581 │ 0.0556098 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       5 │   1100010011 │ 11000 │ 10011 │    24 │    19 │ -2.67742 │ 9.22581 │ 0.0556098 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       1 │   1011011001 │ 10110 │ 11001 │    22 │    25 │ -2.87097 │ 9.6129  │ 0.0486594 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       2 │   1011011001 │ 10110 │ 11001 │    22 │    25 │ -2.87097 │ 9.6129  │ 0.0486594 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       7 │   1100010011 │ 11000 │ 10011 │    24 │    19 │ -2.67742 │ 9.22581 │ 0.0556098 │
╘═════════╧══════════════╧═══════╧═══════╧═══════╧═══════╧══════════╧═════════╧═══════════╛
Crossover list
 [[0 1]
 [2 4]
 [3 5]
 [6 7]]
prob 0.5670111114833867 < probcrossover of 0.9. Perform crossover
crossover site:  3
prob 0.4863112648555784 < probcrossover of 0.9. Perform crossover
crossover site:  4
prob 0.647616732746765 < probcrossover of 0.9. Perform crossover
crossover site:  7
prob 0.07786655138434917 < probcrossover of 0.9. Perform crossover
crossover site:  9
CrossoverTable
╒═════════╤══════════════╤═══════╤═══════╤═══════╤═══════╤══════════╤═════════╤═══════════╕
│   Index │   population │   x1c │   x2c │   x1d │   x2d │       x1 │ x2      │    objval │
╞═════════╪══════════════╪═══════╪═══════╪═══════╪═══════╪══════════╪═════════╪═══════════╡
│       0 │   1011011001 │ 10110 │ 11001 │    22 │    25 │ -2.87097 │ 9.6129  │ 0.0486594 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       1 │   1011011001 │ 10110 │ 11001 │    22 │    25 │ -2.87097 │ 9.6129  │ 0.0486594 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       2 │   1011010011 │ 10110 │ 10011 │    22 │    19 │ -2.87097 │ 9.22581 │ 0.0562685 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       4 │   1100011001 │ 11000 │ 11001 │    24 │    25 │ -2.67742 │ 9.6129  │ 0.0655577 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       3 │   1100010001 │ 11000 │ 10001 │    24 │    17 │ -2.67742 │ 9.09677 │ 0.0546341 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       5 │   1011011011 │ 10110 │ 11011 │    22 │    27 │ -2.87097 │ 9.74194 │ 0.0517413 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       6 │   1011011001 │ 10110 │ 11001 │    22 │    25 │ -2.87097 │ 9.6129  │ 0.0486594 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       7 │   1100010011 │ 11000 │ 10011 │    24 │    19 │ -2.67742 │ 9.22581 │ 0.0556098 │
╘═════════╧══════════════╧═══════╧═══════╧═══════╧═══════╧══════════╧═════════╧═══════════╛
prob 0.10837739121430079 > probmutation of 0.1. DO NOT Perform Mutation
prob 0.28046492916687726 > probmutation of 0.1. DO NOT Perform Mutation
prob 0.9037184024969827 > probmutation of 0.1. DO NOT Perform Mutation
prob 0.23152152157671602 > probmutation of 0.1. DO NOT Perform Mutation
prob 0.2527698191729627 > probmutation of 0.1. DO NOT Perform Mutation
prob 0.7710570612819361 > probmutation of 0.1. DO NOT Perform Mutation
prob 0.7171734250997963 > probmutation of 0.1. DO NOT Perform Mutation
prob 0.7078839758615079 > probmutation of 0.1. DO NOT Perform Mutation
MutationTable
╒═════════╤══════════════╤═══════╤═══════╤═══════╤═══════╤══════════╤═════════╤═══════════╕
│   Index │   population │   x1c │   x2c │   x1d │   x2d │       x1 │ x2      │    objval │
╞═════════╪══════════════╪═══════╪═══════╪═══════╪═══════╪══════════╪═════════╪═══════════╡
│       0 │   1011011001 │ 10110 │ 11001 │    22 │    25 │ -2.87097 │ 9.6129  │ 0.0486594 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       1 │   1011011001 │ 10110 │ 11001 │    22 │    25 │ -2.87097 │ 9.6129  │ 0.0486594 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       2 │   1011010011 │ 10110 │ 10011 │    22 │    19 │ -2.87097 │ 9.22581 │ 0.0562685 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       3 │   1100011001 │ 11000 │ 11001 │    24 │    25 │ -2.67742 │ 9.6129  │ 0.0655577 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       4 │   1100010001 │ 11000 │ 10001 │    24 │    17 │ -2.67742 │ 9.09677 │ 0.0546341 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       5 │   1011011011 │ 10110 │ 11011 │    22 │    27 │ -2.87097 │ 9.74194 │ 0.0517413 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       6 │   1011011001 │ 10110 │ 11001 │    22 │    25 │ -2.87097 │ 9.6129  │ 0.0486594 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       7 │   1100010011 │ 11000 │ 10011 │    24 │    19 │ -2.67742 │ 9.22581 │ 0.0556098 │
╘═════════╧══════════════╧═══════╧═══════╧═══════╧═══════╧══════════╧═════════╧═══════════╛
SurvivalTable
╒═════════╤══════════════╤═══════╤═══════╤═══════╤═══════╤══════════╤═════════╤═══════════╕
│   Index │   population │   x1c │   x2c │   x1d │   x2d │       x1 │ x2      │    objval │
```

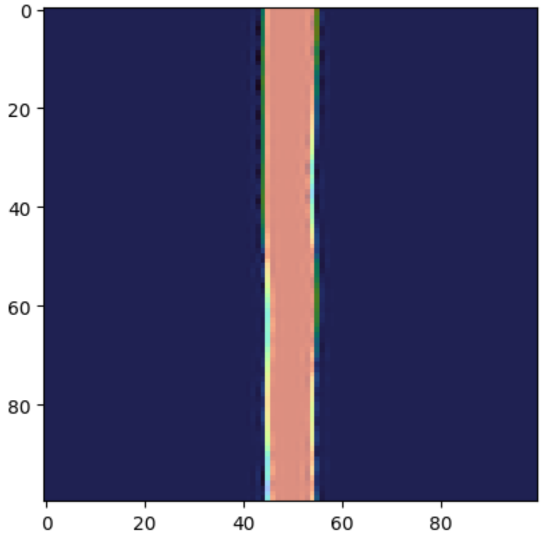
```
|   0 | 1011011001 | 10110 | 11001 |  22 |  25 | -2.87097 | 9.6129  | 0.0486594 |
|   1 | 1011011001 | 10110 | 11001 |  22 |  25 | -2.87097 | 9.6129  | 0.0486594 |
|   6 | 1011011001 | 10110 | 11001 |  22 |  25 | -2.87097 | 9.6129  | 0.0486594 |
|   0 | 1011011001 | 10110 | 11001 |  22 |  25 | -2.87097 | 9.6129  | 0.0486594 |
|   1 | 1011011001 | 10110 | 11001 |  22 |  25 | -2.87097 | 9.6129  | 0.0486594 |
|   2 | 1011011001 | 10110 | 11001 |  22 |  25 | -2.87097 | 9.6129  | 0.0486594 |
|   5 | 1011011011 | 10110 | 11011 |  22 |  27 | -2.87097 | 9.74194 | 0.0517413 |
|   3 | 1011011011 | 10110 | 11011 |  22 |  27 | -2.87097 | 9.74194 | 0.0517413 |
Final result
```

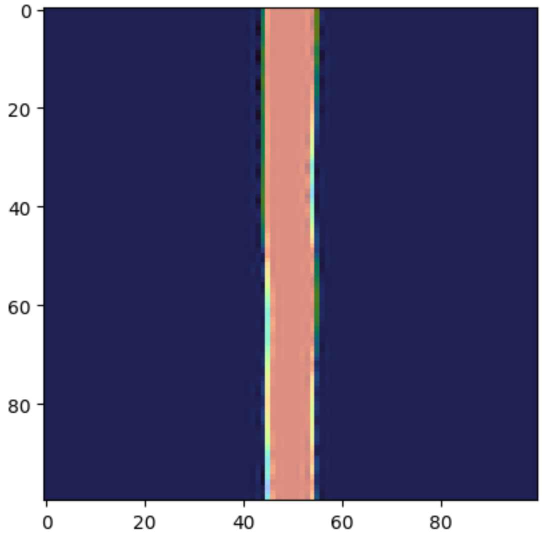


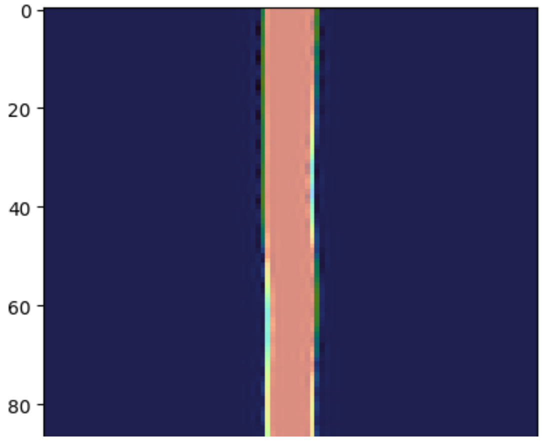


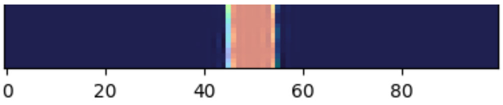


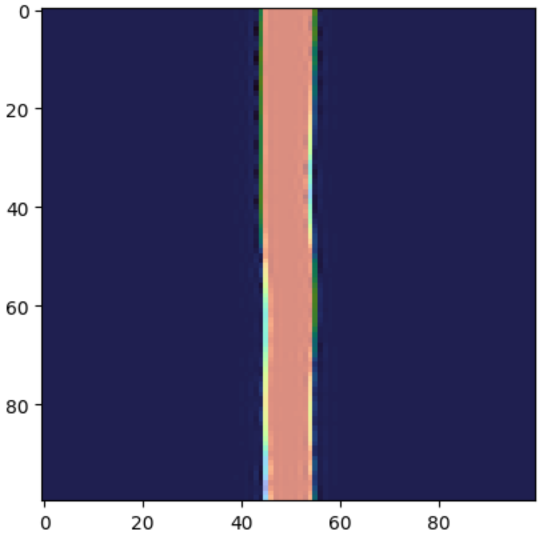

0
20
40
60
80
0
20
40
60
80


0
20
40
60
80
0
20
40
60
80


0
20
40
60
80
0
20
40
60
80


0
20
40
60
80





```
╒═════════╤══════════════╤═══════╤═══════╤═══════╤═══════╤══════════╤═════════╤═══════════╕
│   Index │   population │   x1c │   x2c │   x1d │   x2d │       x1 │      x2 │    objval │
╞═════════╪══════════════╪═══════╪═══════╪═══════╪═══════╪══════════╪═════════╪═══════════╡
│       0 │   1011011001 │ 10110 │ 11001 │    22 │    25 │ -2.87097 │ 9.6129  │ 0.0486594 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       1 │   1011011001 │ 10110 │ 11001 │    22 │    25 │ -2.87097 │ 9.6129  │ 0.0486594 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       2 │   1011011001 │ 10110 │ 11001 │    22 │    25 │ -2.87097 │ 9.6129  │ 0.0486594 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       3 │   1011011001 │ 10110 │ 11001 │    22 │    25 │ -2.87097 │ 9.6129  │ 0.0486594 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       4 │   1011011001 │ 10110 │ 11001 │    22 │    25 │ -2.87097 │ 9.6129  │ 0.0486594 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       5 │   1011011001 │ 10110 │ 11001 │    22 │    25 │ -2.87097 │ 9.6129  │ 0.0486594 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       6 │   1011011011 │ 10110 │ 11011 │    22 │    27 │ -2.87097 │ 9.74194 │ 0.0517413 │
├─────────┼──────────────┼───────┼───────┼───────┼───────┼──────────┼─────────┼───────────┤
│       7 │   1011011011 │ 10110 │ 11011 │    22 │    27 │ -2.87097 │ 9.74194 │ 0.0517413 │
╘═════════╧══════════════╧═══════╧═══════╧═══════╧═══════╧══════════╧═════════╧═══════════╛
```